



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/588,492	06/06/2000	William G. Tuel JR.	POU919990100US1	9151

7590 03/30/2004

Floyd A Gonzalez Attorney  
IBM Corporation Intellectual Property Law Dept  
2455 South Rd M/s P386  
Poughkeepsie, NY 12601

EXAMINER

BANANKHAH, MAJID A

ART UNIT	PAPER NUMBER
----------	--------------

2127

DATE MAILED: 03/30/2004

3

Please find below and/or attached an Office communication concerning this application or proceeding.

h

**Office Action Summary**

Application No.

09/588,492

Applicant(s)

TUEL ET AL.

Examiner

Majid A Banankhah

Art Unit

2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 06 June 2000.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-48 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-48 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

Art Unit: 2127

## DETAILED ACTION

1. This office action is in response to application filed on April 25, 2000. Claims 1-35 are considered for examination.

### *Claim Rejections - 35 USC § 103*

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claim 1-48 are rejected under 35 U.S.C. 35 U.S.C. 103(a) as being unpatentable over Huff et al. (U.S. Pat., No. 6,457,064) in view of UNIX Internals (The New Frontiers, Uresh Vahalia, EMC Corporation Hopkinton, MA, 1996).

Per claims, 1, 13, 25, 37 Huff teaches:

a method for efficiently dispatching threads awaiting messages in a multi-threaded communication library comprising (col.3, lines 7-10, *In another aspect of the invention, a method of invoking a thread in a process when an input event is received and using a reduced number of light weight processes is disclosed*): preassigning threads to messages to be received (col. 5, lines 1-5, *In the described embodiment, a child process, a thread, and a connection associated with that thread each have a number or identifier. This information is stored in a shared memory 116, having a series of data cells 118, that can be read and updated by all child processes*); those threads whose assigned messages have not been received are in wait state (col. 6, lines 12-20, *In a typical connection, a thread is generally waiting for input or some type of activity to occur the majority of the time the thread is in existence. In the described embodiment, the input wait table for a process stores information on which connections, i.e., threads, are in a wait state and which are executing or processing an input event. In the described embodiment, the operating system assigns to the process a light weight process for maintaining the process's input wait table*); upon receipt of a message, awakening its preassigned thread; and executing said awakened thread, thereby processing the received message (col. 6, lines 30-37, *the associated with the input wait table invokes a special thread referred to as a polling thread when an input event is received. The polling thread polls or scans the table to determine which connection or connections the input event is directed to and changes the state of the connection from the wait state to a state indicating that it has received input that needs to be processed*).

Art Unit: 2127

The reference of Kakuta while teaches of threads that have not received the message are being in the wait state, fails to specifically teaches of putting to sleep those whose assigned messages have not been received. However, it is notoriously known in the art that threads that are not eligible to run are typically blocked [put in sleep state], for the reason to maximizing the efficiency of multithreading system as it is evidenced by UNIX Internals (See page 53, section 3.2.1 Kernel Threads, lines 1-9, standard synchronization mechanism of the kernel. Therefore, it would have been obvious for one ordinary skill in the art at the time the invention was made to put the waiting threads to sleep before they receive their messages, for the purpose of increasing the efficiency of the multithreading system by using standard synchronization.

Per claims 2, 14, 26, and 38 wherein the selection of the thread to be dispatched is based on its priority as set when the thread is put to sleep, the limitation is taught by UNIX Internals in page 63, last paragraph (*the thread scheduler selects a thread from the queue based on priority*)

Per claims 3, 15, 27, and 39 creating a thread-specific structure for each thread, each thread-specific structure having a ready flag and a condition variable unique to its preassigned thread, creating a handle for each message to be received; and having a thread invoke message passing logic for a particular handle, thereby associating the thread and the message, the reference of Huff teaches of a thread specific structure a condition unique to a preassigned thread (col. 5, lines 16-35, *Once a thread is created within a child process, a thread specific data cell 118 is assigned to that thread. In the described embodiment, this shared memory 116 is created and pre-allocated by the parent process when the server is activated. In other preferred embodiments, the shared memory 116, if needed, can be created by other entities in the operating system. As mentioned, the shared memory is made up of a series of data cells. These cells 118 are identified by a coordinate "i" corresponding to a process and a coordinate "j" corresponding to a thread within that process. Thus, cell (P.sub.i, T.sub.j) is a thread specific data cell, which also contains a connection number "k," that allows one thread to inform other threads of its actions, such as updating a mailbox or copying messages from a mailbox. The thread-specific data cells 118 of the described embodiment in shared memory 116 allow a thread to inform all other threads under the same parent process of that thread's actions. Thus, the shared memory resides on the server and is pre-allocated and controlled by a parent process once the parent process is invoked*).

Per claims 4, 16, 28, and 40 enqueueing for a received message, a preassigned thread-specific structure into a first queue; writing into said handle associated with the message received, an identification of said thread-specific structure enqueued for the received message, and placing said thread-specific structure for the received message in the WAIT condition (col. 6, lines 12-20, *In a typical connection, a thread is generally waiting for input or some type of activity to occur the majority of the time the thread is in existence. In the described embodiment, the input wait table for a process stores information on which connections, i.e., threads, is in a wait state and which are executing or processing an input event. In the described embodiment, the operating system assigns to the process a lightweight process for maintaining the process's input wait table*).

Art Unit: 2127

Per claims 5-6, 17-18, 29-30, and 41-42 the specifics of the thread states is explained by Huff in col. 7, lines 17-37, and 59-68, continued on col. 8, lines 1-2.

Per claims 7, 19, 31, and 43 storing said received message in the buffer identified in the associated handle for the received message (col. 4, lines 30-38, *For example, in the described embodiment, a thread represents a client connection. Following this example, the client connection can be to a mail message store on a server in which the user wants to access mail messages. An active connection in this context represents a user session to a message store in a large network. In this environment, a parent process receives user requests to access a mail message store manages several child processes, each containing typically many active connection threads. This configuration of the described embodiment is shown in FIG. 1).*

Per claims 8-9, 20-21, 32-33, and 44-45 dequeuing thread from a queue according to FIFO or LIFO is well known in the art and by definition a queue is a data structure from which elements can be removed only in the same order in which they are inserted: that is it follows a FIFO or LIFO constraint, therefore.

Per claims 10-11, 22-23, 34-35, and 46-47 dequeuing thread from a queue according to priority is well known in the art and by definition a queue is a data structure from which elements can be removed based on FIFO, LIFO or priority assigned to the elements.

Per claims 12, 24, 36, and 48 further comprising obtaining a lock for the handle associated with said received message such that the awakened thread may process only the received message (See, Unix Internals, page 56, last paragraph).

4. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to **Majid A. Banankhah** whose voice telephone number is **(703) 308-6903**. A voice mail service is also available at this number.

All response sent to U.S. Mail should be mailed to:

**Commissioner of Patent and Trademarks  
Washington, D.C. 20231**

**Hand-delivered responses should be brought to Crystal Park Two, 2021 Crystal Drive, Arlington, VA, Six Floor (Receptionist).** All hand-delivered responses will be handled and entered by the docketing personnel. Please do not hand deliver responses to the Examiner.

**All Formal or Official Faxes must be signed and sent to either (703) 308-9051 or (703) 308-9052.** Official faxes will be handled and entered by the docketing personnel. The date of entry will correspond to the actual FAX reception date unless that date is a Saturday, Sunday, or a Federal Holiday within the District of Columbia, in which case the official date of receipt will be the next business day. The application file will be promptly forwarded to the Examiner

Art Unit: 2127

unless the application file must be sent to another area of the office, e.g., Finance Division for fee charging, etc.

Any inquiry of a general nature or relating to the status of this application should be directed to the Group receptionist whose telephone number is (703) 305-9600.

Majid Banankhah

3/21/04

  
MAJID BANANKHAH  
PRIMARY EXAMINER